

## METHOD FOR AUTOMATIC TUNNELING OF IPv6 PACKETS WITH TOPOLOGICALLY INCORRECT SOURCE ADDRESSES

### FIELD OF THE INVENTION

[0001] The present invention relates to apparatuses, methods and a computer program product for providing Method for automatic tunneling of IPv6 packets with topologically incorrect source addresses.

### RELATED BACKGROUND ART

[0002] The following meanings for the abbreviations used in this specification apply:

[0003] GPRS General Packet Radio Service.

[0004] GTP GPRS Tunneling Protocol

[0005] IANA Internet Assigned Numbers Authority

[0006] IETF Internet Engineering Task Force

[0007] IPv6 Internet Protocol version 6

[0008] LMA Local Mobility Anchor

[0009] MAC Media access control

[0010] MAG Mobile Access Gateway

[0011] MN Mobile Node

[0012] PMIP6 Proxy Mobile IPv6

[0013] RA Router Advertisement

[0014] Embodiments of the present invention relate to IPv6 addressing in the case of network based mobility framework, when local IPv6 address prefixes are used, and for numbering of very resource constrained nodes, such as battery, memory, and processing power constrained.

[0015] A resource constrained node is a device that has less CPU power, battery power, ROM or RAM memory, or resources alike, than what is considered to be common for Internet connected devices. For resource constrained nodes saving of all resources is of extreme importance.

[0016] The idea of using local IPv6 prefixes, for offloading and local access when network based mobility is used, is topical.

[0017] What comes to resource constrained nodes; the currently available addressing methods are essentially automated address configuration tools, such as Stateless Address Autoconfiguration and Dynamic Host Configuration Protocol version 6. These both are quite complex for resource constrained nodes. Manual IPv6 address configuration is theoretically possible, but usually not in practice even with PCs, let alone with these resource constrained nodes.

[0018] A similar problem may occur when a mobile node (MN) using PMIP6 and having a local address in an old network moves to a new network, wherein in the new network, a new address would be necessary, wherein at least during the address re-configuration, connection loss may occur.

[0019] Thus, at present there is no satisfying way of handling addresses in particular of resource constrained nodes (such as machine type communication (MTC) nodes, for example).

### SUMMARY OF THE INVENTION

[0020] Embodiments of the present invention address these situations and aim to provide a simple solution of handling addresses of resource constrained nodes and to provide a solution for providing mobility support for local IPv6 addresses.

[0021] According to a first aspect of the present invention an apparatus is provided which comprises a first interface unit configured to provide connection to a first network; a second interface unit configured to provide connection to a network node located in a second network; a processor configured to carry out a gateway function between the first network and the second network, to receive a packet from the network node via the second interface unit, wherein the packet comprises a source address which topologically does not belong to the second network, to encapsulate the received packet in a new packet, and to send the new packet to the first network via the first interface unit.

[0022] According to a second aspect of the present invention an apparatus is provided which comprises a first interface unit configured to provide connection to a first network; a second interface unit configured to provide connection to a second network; and a processor configured to carry out a gateway function between the first network and the second network, to receive an encapsulated packet from the first network via the first interface unit, to decapsulate the received packet, wherein the decapsulated packet comprises a destination address which topologically does not belong to the second network, and to send the decapsulated packet to the destination address to the second network via the second interface unit.

[0023] According to a third aspect of the present invention an apparatus is provided which comprises an interface unit configured to provide connection to a first network; and a processor configured to receive a packet from a gateway via the first network via the interface unit, wherein the packet is encapsulated and the gateway is located between the first network and a second network, to record the source address of the packet as the address of the gateway, to decapsulate the received packet, and to record the source address of the decapsulated packet as an address of the network node located in the second network, wherein the address of the network node does topologically not belong to the second network.

[0024] According to a fourth aspect of the present invention an apparatus is provided which comprises an interface unit configured to provide connection to a first network; and a processor configured to create a packet having an address of a network node as a destination address, wherein the network node is located in a second network and the address of the network node does topologically not belong to the second network, to encapsulate the created packet into an encapsulated packet having the address of a gateway between the first network and the second network, and to send the encapsulated packet to the gateway.

[0025] According to a fifth aspect of the present invention a method is provided which comprises

[0026] carrying out a gateway function between a first network and a second network,

[0027] receiving a packet from a network node located in the second network, wherein the packet comprises a source address which topologically does not belong to the second network,

[0028] encapsulating the received packet in a new packet, and

[0029] sending the new packet to the first network.

[0030] According to a sixth aspect of the present invention a method is provided which comprises

[0031] carrying out a gateway function between a first network and a second network,